```
clear;
%%%% %%%%%%%%%%%% Authors-cum-Engineers %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

MWONGELA D MATHINA: F19/1707/2013

WANJALA N KOTOCHAI: F19/1717/2013

```
%%%%%%%%%%%%%%%%% we get going %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%% photo coordinates of photo 1 %%%%%%%%%%
photo1=[0.966 -88.738
        -0.798 1.403
        -2.511 92.055
        92.337 -88.145
        96.602 3.491
        85.136 90.647];
%%%%%%%%%%% photo coordinates of photo 2 %%%%%%%%%%
photo2=[-91.627 -86.419
        -89.994 4.162
        -88.824 95.641
        -1.022 -89.392
        0.818 2.564
        2.595 90.518];

 %%%%%%%% model coordinates of the 6 conugate image points %%%%%%%%%%%%
model=[0 0 -152
       200 0 -152
       0 187.5 -152
       200 187.5 -152
       0 -187.5 -152
       200 -187.5 -152];
X=model (:,1);
Y=model(:,2);
ZA=model(:,3);
Z=(200/92)*ZA;

    f=152; %%%%% focal length %%%%
 %%%%extracting photo coordinates %%%%
xa1=photo1(:,1);
ya1=photo1(:,2);
xa2=photo2(:,1);
ya2=photo2(:,2);

%%%% initial values %%%%%
om=0;phi=0;K=0;
 BY=0;BZ=0;BX=200;

 %%%%%%%%%% column matrices to aid in formation of the A matrix %%%%%%
 M=[1;1;1;1;2;2;2;2;3;3;3;3;4;4;4;4;5;5;5;5;6;6;6;6];%%%%%% column matrix helps pi
 T=[1;1;1;1;3;3;3;3;5;5;5;5;7;7;7;7;9;9;9;9;11;11;11;11];%%%%%% column matrix plac

 %%%%%%%%%%%% Functions of the three rotations omega,phi and kappa %%%%%
```

```
 r11=cos(phi)*cos(K);
 r12=((cos(om)*sin(K))+(sin(om)*sin(phi)*cos(K)));
 r13=(sin(om)*sin(K))-(cos(om)*sin(phi)*cos(K));
 r21=-cos(phi)*sin(K);
 r22=(cos(om)*cos(K))-(sin(om)*sin(phi)*sin(K));
 r23=(sin(om)*cos(K))+(cos(om)*sin(phi)*sin(K));
 r31=sin(phi);
 r32=-sin(om)*cos(phi);
 r33=cos(om)*cos(phi);

for i=1:100 %%%%%% iterations loop %%%%%

 %%%%%% for loop to form the A matrix %%%%%%
 for i=1:24;
     m=M(i);
     k=T(i);
     i=2*k-1;
     j=3*m;

  %%%%%%%%%%%%%%%%%% differential matrix w.r.t omega  %%%%%%%%

     A(i,1)=0;
     A(i+1,1)=0;
     dom1=((-ya2(m)*sin(om)*sin(K)+ya2(m)*cos(om)*sin(phi)*cos(K)-f*cos(om)*sin(K)
     dom2=((-ya2(m)*cos(om)*cos(phi)+f*sin(om)*cos(phi))*(X(m)-200));
     A(i+2,1)=dom1-dom2;
     dom3=((-ya2(m)*sin(om)*cos(K)-ya2(m)*cos(om)*sin(phi)*sin(K))-(f*cos(om)*cos(
     dom4=(-ya2(m)*cos(om)*cos(phi)+f*sin(om)*cos(phi))*(Y(m)-BY);
     A(i+3,1)=dom3-dom4;


   %%%%%%%%%%%%%%%%%% differential matrix w.r.t phi  %%%%%%%%

     A(i,2)=0;
     A(i+1,2)=0;
     dphi1=(-xa2(m)*sin(phi)*cos(K)+ya2(m)*sin(om)*cos(phi)*cos(K)+f*cos(om)*cos(p
     dphi2=(xa2(m)*cos(phi)+ya2(m)*sin(om)*sin(phi)+f*cos(om)*sin(phi))*(X(m)-200)
     A(i+2,2)=dphi1-dphi2;
     dphi3=(xa2(m)*sin(phi)*sin(K)-ya2(m)*sin(om)*cos(phi)*sin(K)-f*cos(om)*cos(ph
     dphi4=(xa2(m)*cos(phi)-ya2(m)*sin(om)*sin(phi)+f*cos(om)*sin(phi))*(Y(m)-BY);
     A(i+3,2)=dphi3-dphi4;


    %%%%%%%%%%%%%%%%%% differential matrix w.r.t kappa  %%%%%%%%

     A(i,3)=0;
     A(i+1,3)=0;
     A(i+2,3)=(-xa2(m)*cos(phi)*sin(K)+ya2(m)*cos(om)*cos(K)-ya2(m)*sin(om)*sin(ph
     A(i+3,3)=(-xa2(m)*cos(phi)*cos(K)-ya2(m)*cos(om)*sin(K)-ya2(m)*sin(om)*sin(ph

    %%%%%%%%%%%%%%%%%% differential matrix w.r.t base component BY  %%%%%%%%

     A(i,4)=0;
     A(i+1,4)=0;
```

```
      A(i+2,4)=0;
      A(i+3,4)=r31*xa2(m)+r32*ya2(m)-r33*f;


    %%%%%%%%%%%%%%%%% differential matrix w.r.t base component BZ  %%%%%%%%

      A(i,5)=0;
      A(i+1,5)=0;
      A(i+2,5)=-(r11*xa2(m)+r12*ya2(m)-r13*f);
      A(i+3,5)=-(r21*xa2(m)+r22*ya2(m)-r23*f);

     %%%%%%%%%%%%%%%%% differential matrices w.r.t model coordinates X,Y,Z %%%%%%%%

     %%%%%%%%% matrix as a result of X %%%%%%%
      A(i,3+j)=f;
      A(i+1,3+j)=0;
      A(i+2,3+j)=-(r31*xa2(m)+r32*ya2(m)-r33*f);
      A(i+3,3+j)=0;

     %%%%%%%%% matrix as a result of Y %%%%%%%

      A(i,4+j)=0;
      A(i+1,4+j)=f;
      A(i+2,4+j)=0;
      A(i+3,4+j)= -(r31*xa2(m)+r32*ya2(m)-r33*f);

      %%%%%%%%% matrix as a result of Z %%%%%%%

      A(i,5+j)=xa1(m);
      A(i+1,5+j)=ya1(m);
      A(i+2,5+j)=r11*xa2(m)+r12*ya2(m)-r13*f;
      A(i+3,5+j)=r21*xa2(m)+r22*ya2(m)-r23*f;
   end

   %%%%%%Formation of the matrix of constants;the L matrix %%%%%%%

  for i=1:24;
      c=M(i);
      g=T(i);
      i=2*g-1;
      L(i,1)=xa1(c)*Z(c)+f*X(c);
      L(i+1,1)=ya1(c)*Z(c)+f*Y(c);
      L(i+2,1)=((r11*xa2(c)+r12*ya2(c)-r13*f)*(Z(c)-BZ))-((r31*xa2(c)+r32*ya2(c)-r3
      L(i+3,1)=((r21*xa2(c)+r22*ya2(c)-r23*f)*(Z(c)-BZ))-((r31*xa2(c)+r32*ya2(c)-r3
  end

  N=A'*A; %%%%%Normal equation matrix %%%%%%%
Qxx=inv(N);%%%%%%%% coffactor matrix %%%%%%
d=A'*-L;   %%%%% absolute vector %%%%%%%%
delta=Qxx*d; %%%%% corrections %%%%%%%%

%%%%updating the initial values %%%%%%

om=om+delta(1,1);
```

```
phi=phi+delta(2,1);
K=K+delta(3,1);
BY=BY+delta(4,1);
BZ=BZ+delta(5,1);
X(1,1)=X(1,1)+delta(6,1);
Y(1,1)=Y(1,1)+delta(7,1);
Z(1,1)=Z(1,1)+delta(8,1);
X(2,1)=X(2,1)+delta(9,1);
Y(2,1)=Y(2,1)+delta(10,1);
Z(2,1)=Z(2,1)+delta(11,1);
X(3,1)=X(3,1)+delta(12,1);
Y(3,1)=Y(3,1)+delta(13,1);
Z(3,1)=Z(3,1)+delta(14,1);
X(4,1)=X(4,1)+delta(15,1);
Y(4,1)=Y(4,1)+delta(16,1);
Z(4,1)=Z(4,1)+delta(17,1);
X(5,1)=X(5,1)+delta(18,1);
Y(5,1)=Y(5,1)+delta(19,1);
Z(5,1)=Z(5,1)+delta(20,1);
X(6,1)=X(6,1)+delta(21,1);
Y(6,1)=Y(6,1)+delta(22,1);
Z(6,1)=Z(6,1)+delta(23,1);

end  %%%% end of iteration loop %%%%%%
%%%%%%%%%% relative orientation parameters and the model coordinates %%%
R_parameters_modelcoords=[om;phi;K;BY;BZ;X(1,1);Y(1,1);Z(1,1);X(2,1);Y(2,1);Z(2,1)

%%%%%%% Accuracy assessment %%%%%%%
%%%% v=L-Ax %%%
```

```
R_parameters_modelcoords =

    0.3981
    0.0583
   -2.7210
    1.9855
   -8.9893
    2.2379
 -205.6113
 -352.1757
   -1.8552
    3.2807
 -353.4778
   -5.8263
  213.5754
 -352.6660
  210.8758
 -201.2892
 -347.1323
  207.0082
    7.4562
 -325.7200
  204.5202
```

```
  217.7707
 -365.1460
```

```
Ax=A*delta;
v=L-Ax; %%%% Residual vector %%%%
sigma=sqrt((v'*v)/(24-23));%%%%% aposteriori variance %%%%%%%%%
Exx=sigma*Qxx; %%%%%%% covariance matrix %%%%%%
 E1 = diag(Exx);%%%%% extracting the diagonal matrix to compute std deviations %%%
 S_deviations=sqrt(E1)%%%%%%%%% Standard deviations %%%%%%
```

```
S_deviations =

    0.0068
    0.0061
    0.0035
    1.6777
    2.0471
    0.5548
    2.0043
    2.9795
    0.5657
    0.5378
    3.8726
    0.5837
    3.3629
    5.4861
    2.0135
    1.9081
    3.4413
    2.0344
    0.5217
    3.3243
    2.6627
    2.7261
    4.8640
```

Published with MATLAB® R2014a